

Strings

Programación I

Lic. Mauro Gullino maurogullino@gmail.com

UTN FRH

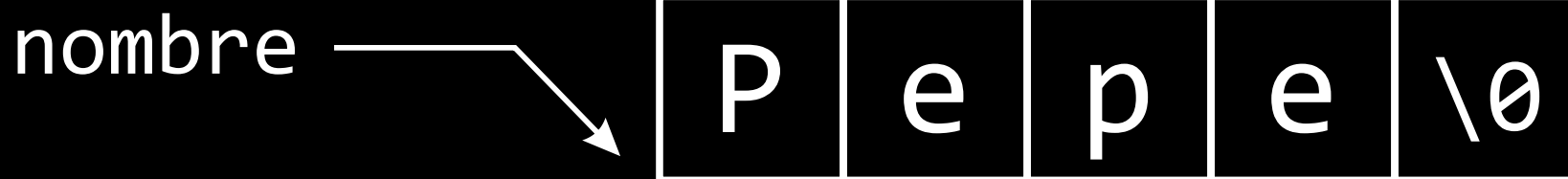
String / cadena

Es un vector de caracteres que finaliza con el **caracter nulo**.

Caracter nulo es un caracter con código ASCII cero (evalúa como falso)

Asignación directa

```
char nombre[5];  
nombre[0] = 'P';  
nombre[1] = 'e';  
nombre[2] = 'p';  
nombre[3] = 'e';  
nombre[4] = '\\0'; //caract. nulo
```



Asignación directa

```
char nombre[] = {'P', 'e', 'p', 'e', '\0'};
```

```
char nombre[] = {80, 101, 112, 101, 0};
```

```
char nombre[] = "Pepe";
```

Asignación directa

```
char nombre[] = "Pepe";
```

!=

```
char nombre[5];  
nombre = "Pepe";
```

!!!

Ejemplo

```
#include <stdio.h>
```

```
main() {
```

```
    char i, prueba[]="Pepe";
```

```
    for(i=0; i<5; i++)
```

```
        printf("\n %d %c", prueba[i], prueba[i]);
```

```
}
```

Ingreso por teclado

```
char palabra[20];  
scanf("%s", palabra);
```

```
char palabra2[20];  
gets(palabra2);
```



dirección

Salida a pantalla

```
char frase[] = "Hola mundo cruel";  
puts(frase);  
puts("Adios mundo cruel");
```

```
printf("%s", frase);
```



dirección

Biblioteca

string.h

strcmp0

Recibe dos strings y retorna la "comparación"

Valor retornado:

`== 0` son iguales

`> 0` string1 mayor a string2

`< 0` string1 menor a string2

```
int comp;
```

```
comp = strcmp("mano", "mono");
```

strlen()

Recibe una string y devuelve su "largo" (sin contar el nulo)

```
int tamano;  
tamano = strlen("mundo");
```

strcpy0

Recibe dos direcciones de memoria y copia los char de una en otra, hasta encontrar un nulo, que también se copia.

```
#define TAM 20
```

```
char palabra[TAM], sinonimo[TAM];  
strcpy(sinonimo, "Probando");  
strcpy(palabra, sinonimo);
```



Ejercicio

Realice un programa que reciba palabras hasta que se ingrese "FIN".

Se debe informar cuál fue la palabra más extensa ingresada.

Errores comunes

```
char cadena[10];  
cadena = "Pepe";
```

```
// asignación ilegal  
// lo correcto sería:
```

```
strcpy(cadena, "Pepe");
```

Errores comunes

```
char cadena[]="Pepe";  
char cadena2[10];
```

```
cadena2 = cadena;
```

```
// copia ilegal
```

```
// lo correcto sería:
```

```
strcpy(cadena2, cadena);
```

Errores comunes

```
char cadena[] = "Argentina";  
char cadena2[] = "Argentina";
```

```
if (cadena == cadena2) {    //no!!!  
    printf("Son iguales");  
} else {  
    printf("Son distintas");  
}
```


Errores comunes

```
char cadena[] = "Argentina";  
char cadena2[] = "Argentina";
```

```
if (strcmp(cadena,cadena2) == 0 ) { //sí  
    printf("Son iguales");  
} else {  
    printf("Son distintas");  
}
```

Vectores de strings

¿Cómo podríamos administrar una lista de palabras?

¿Un vector de strings?

Dado que una string es un vector, un vector de string es lo mismo que decir una matriz de char.

Vector de strings

```
#define CANTCIU 5
#define LETRASMAX 30
char ciudades[CANTCIU][LETRASMAX];

strcpy( ciudades[0], "Buenos Aires");
strcpy( ciudades[1], "Caracas");

printf("%c", ciudades[0][2] );
```

Ejemplo

Se ingresan por teclado cuatro ciudades.

Luego se ingresa una ciudad a buscar y el programa debe informar si la ciudad se encuentra entre las ingresadas.

Ejemplo

Se ingresan por teclado cuatro ciudades.

El programa debe mostrarlas ordenadas alfabéticamente.

Ejemplo

Se ingresan por teclado cuatro ciudades y su población en millones de habitantes.

El programa debe mostrarlas ordenadas por población.